

En este ejercicio vamos a seguir con el desarrollo de la agenda. Vamos a construir la pantalla inicial que será un recyclerView. En un principio la agenda no estará ordenada, las inserciones en la misma son secuenciales (por el final). Para manejar esta lista tendremos un array.



Lo único que tenemos que implementar, por ahora, es el control de las pulsaciones sobre los elementos del recycler.

Funcionalidad del Click en el Botón flotante

Al pulsar sobre el FAB se mostrará un Toast indicando que se añadirá un nuevo contacto. Se dejará la implementación para temas posteriores.



Funcionalidad del Click

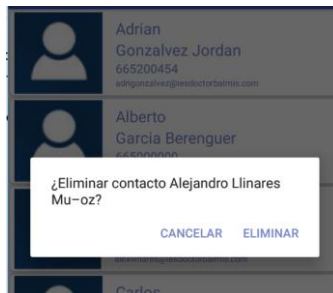
En caso de realizar una pulsación corta, OnClick, se mostrará un Toast indicando el nombre del contacto a editar.



Funcionalidad del LongClick

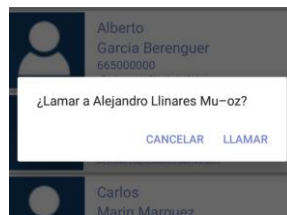
En caso que el usuario realice una pulsación larga, OnLongClick, sobre algún, se mostrará un dialogo preguntando si se quiere eliminar el contacto. Solo se eliminará si pulsamos a eliminar, en caso contrario se cerrará el dialogo. Eliminaremos el contacto del array y actualizaremos el adaptador.



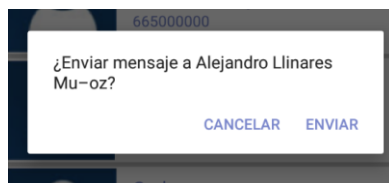


Funcionalidad del desplazamiento izq/derch

Si en la ventana inicial hago un **desplazamiento hacia la derecha** encima de un contacto entonces marco el dial con el teléfono del contacto, mostrando un dialogo que permita aceptar o rechazar lanzar la llamada. Si no tiene teléfono se dará un mensaje de advertencia.



Si en la ventana principal hago un **desplazamiento hacia la izquierda** se mandará un correo al contacto. Si no tiene dirección de correo se presentará un mensaje indicandolo.



Para ayudar a la funcionalidad de los movimientos derecha e izquierda, se adjunta una clase `swipeDetector` que gestionará el tipo de movimiento. Esta clase implementa un *listener* para detectar los movimientos sobre la pantalla. En la `MainActivity` definiremos, una variable de esta clase.

```
//Listener para el onTouch de la lista
SwipeDetector swipeDetector = new SwipeDetector();
```

Y a continuación escribiremos el `setOnItemClickListener()` de nuestro *adaptador* usando los métodos de la clase para detectar los movimientos. Importante, tendremos que pasar el objeto `swipeDetector` al listener correspondiente de la clase adaptador.

```
adaptador.setOnTouch(swipeDetector);
```

Funcionalidad del Click sobre la imagen del contacto.

En caso de pulsar sobre la imagen se mostrará un dialogo personalizado en el que la imagen se mostrará de un tamaño mayor junto con el nombre del contacto, como podemos ver en la siguiente captura:

